VALIDATION, VERIFICATION, AND TESTING PLAN

Resident Assessment Subsystem (RASS)
Release 8.4.0.0

U.S. Department of Housing and Urban Development

Version 1

July 28, 2004

Revision Sheet

Release No.	Date	Revision Description
Rev. 1	7/28/2004	Final Version
Rev. 1.2	7/27/2004	Client Review
Rev. 1.1	7/26/2004	Internal Team Review
Rev. 1.0	7/26/2004	Initial document

Validation, Verification, and Testing Plan Authorization Memorandum

I have carefully assessed the Validation, Verification, and Testing Plan for the Resident Assessment Subsystem (RASS) Release 8.4.0.0. This document has been completed in accordance with the requirements of the HUD System Development Methodology.

MANAGEMENT CERTIFICATION - Please check the appropriate statement.

_____ The document is accepted.

_____ The document is accepted pending the changes noted.

_____ The document is not accepted.

We fully accept the changes as needed improvements and authorize initiation of work to proceed. Based on our authority and judgment, the continued operation of this system is authorized.

_____ _____

Delton Nichols DATE
RASS Project Manager

_____ _____

Yangja Lee DATE
RASS GTM/IT Manager

VALIDATION, VERIFICATION AND TESTING PLAN

TABLE OF CONTENTS

Page #

1.0 GENERAL INFORMATION

1 GENERAL INFORMATION

1.1 Purpose

The RASS Release 8.4.0.0 Department of Housing and Urban Development (HUD) System Development Methodology (SDM) Validation, Verification and Testing (VVT) Plan is the third document in a series of five HUD SDM Design Phase deliverable documents required from the RASS System Design and Development Team. This series of HUD SDM Design Phase deliverables includes the System/Subsystem Specifications; Program Specifications; the Validation, Verification and Testing Plan; Training Plan; and Database Specifications.

This document outlines the RASS Release 8.4.0.0 Validation, Verification and Testing Plan for new and modified RASS functionality. The RASS Release 8.4.0.0 Validation, Verification, and Testing Plan provides guidance for management and technical efforts throughout the System, Integration, and User Acceptance Testing period after the development Unit Testing period has finished. It establishes a comprehensive plan to communicate the nature and extent of the test necessary for a thorough evaluation of the Resident Assessment Subsystem (RASS).

1.2 Scope

RASS Release 8.4.0.0 will expand on RASS Release 8.3.1.0 functionality, and will be implemented by December 17, 2004. The following tables describe the functionality to be implemented.

New Capabilities (High Level)
1. Update the RASS Implementation Plan Activity functionality to ensure that PHAs perform activities to implement the survey close to the defined Implementation Plan window.
2. Create an At-risk PHA report to pull up a list of PHAs for a selected field office that are at risk of failing RASS.
3. Enhance the scoring process to provide the capability to automatically assign last year's survey scores for a selected PHA.

Upgraded Capability (High Level)
1. Update reports to support the changes to the PHAS regulations such as the Small PHA Deregulation Rule and FYE changes.
2. Expand the Demographic report to display PHA level survey section scores by a resident's age and gender.
3. Enhance the PHA History Report to display the total number of units, the number of low rent units, the number of Section 8 units, the number of occupied units for a PHA and update the Follow-up Plan certification information column to display "Not Required" if the Follow-Up Plan certification is not required.

Elimination of Deficiencies
1. Standardize the RASS Component access dates by implementing the same logic for inserting the certification window dates and the logic for verification against dates for all three components.

2. Fix the undeliverable address count calculation on the Required Survey Size Threshold Report to only display the counts for selected FYE.
3. Fix the Comment functionality on the Review/Approve PHA Scores report to work with special characters.

Elimination/Reduction of Capabilities No Longer Needed
1. Update the RASS online system to remove the consortia functionality as the calculation of a consortia PHA's score is being moved outside the PIH-REAC's Secure Systems.

1.3 System Overview

The following table identifies the system environment for RASS Release 8.4.0.0.

System Environment	
System	Real Estate Assessment Center System (REACS)
Subsystem	Resident Assessment Subsystem (RASS).
Sponsor	Public and Indian Housing – Real Estate Assessment Center (REAC)
PCAS	00307680
System Code	P089
System Category	Non-major
Operational Status	Operational
System Environment	Web Based
Requirements	Quality Software Services, Inc. (QSSI - Functional and System Requirements)
Design	QSSI
Development	QSSI
System and Integration Testing	QSSI
User Acceptance Testing	HUD RASS Client Team
Deployment	QSSI with REAC DCG (Paradigm/ Booz-Allen & Hamilton)
Maintenance	QSSI

The following table identifies and briefly describes the different users of RASS.

User Environment	
REAC RASS Business Support Team	The REAC RASS Business Support Team uses RASS to view PHA and development-level scoring and certification reports and Multifamily Survey result information. The team also uses the system to complete the various processes (Unit Address Sampling; Implementation Plan; Survey and Follow-up Plan Scoring; RASS Score Approval) required during resident assessment.

User Environment	
Public Housing Agency (PHA)	PHA(s) use RASS to participate annually in the Resident Assessment process. PHA(s) certify their unit addresses, complete an Implementation Plan for resident notification about the annual resident survey process, and complete a Follow-up Plan outlining sources of funding and dates to be completed for required areas (low-scoring survey sections) identified by the resident survey results. PHA(s) may also use the system to view survey result and resident response rate information.
Survey Administrator	After the sampling program has generated a file of units to be surveyed, RASS downloads the sample file to the Survey Administrator. The Survey Administrator distributes the RASS survey to the units indicated in the sample file. Once the Survey Administrator collects and analyzes survey data, the survey results and response rate information are uploaded in RASS from the Survey Administrator so that PHA(s) and Multifamily Owners/Agents may view this information.
Multifamily (MF) Owners/Agents	MF Owners/Agents use RASS to view survey result information for their assigned properties.
Other HUD Users (includes HUD PIH Field Offices)	All other HUD users have read-only access to RASS in order to review PHA and Multifamily Housing development or property level survey result information.

1.4 Project References

The following documents are available to provide a comprehensive understanding of the resident assessment process. Most documents are available via the REAC Document Library. Additionally, several of the documents listed below are available through the PHA Resident Assessment Internet site at ***http://www.hud.gov/offices/reac/products/prodrass.cfm***

Release 8.4.0.0
"RASS Release 8.4.0.0 Needs Statement", QSSI, 06/30/04
"RASS Release 8.4.0.0 Project Plan", QSSI, 06/30/04
"RASS Release 8.4.0.0 Feasibility Study", QSSI, 06/30/04
"RASS Release 8.4.0.0 Cost/Benefit Analysis", QSSI, 06/30/04
"RASS Release 8.4.0.0 System Decision Paper", QSSI, 06/30/04
"RASS Release 8.4.0.0 Risk Analysis", QSSI, 06/30/04
"RASS Release 8.4.0.0 Functional Requirements Document", QSSI, 07/20/04
"RASS Release 8.4.0.0 Data Requirements Document", QSSI, 07/20/04
"RASS Release 8.4.0.0 System Support & Acquisition Plan", QSSI, 07/09/04
"RASS Release 8.4.0.0 System Security & Privacy Plan", QSSI, 07/09/04
"RASS Release 8.4.0.0 Unit Test Plan", QSSI, 07/28/04
"RASS Release 8.4.0.0 System-Subsystem Specification", QSSI, 07/28/04
"RASS Release 8.4.0.0 Validation, Verification and Testing Plan", QSSI, 07/28/04

Release 8.3.0.0

Release 8.3.0.0
"RASS Release 8.3.0.0 Data Requirements Document", QSSI, 12/19/03.
"RASS Release 8.3.0.0 Functional Requirements Document", QSSI, 12/19/03.
"RASS Release 8.3.0.0 System Support & Acquisition Plan", QSSI, 12/19/03.
"RASS Release 8.3.0.0 System Security & Privacy Plan", QSSI, 12/19/03.
"RASS Release 8.3.0.0 Needs Statement", QSSI, 12/09/03.
"RASS Release 8.3.0.0 Project Plan", QSSI, 12/09/03.
"RASS Release 8.3.0.0 Feasibility Study", QSSI, 12/09/03.
"RASS Release 8.3.0.0 Cost/Benefit Analysis", QSSI, 12/09/03.
"RASS Release 8.3.0.0 System Decision Paper", QSSI, 12/09/03.
"RASS Release 8.3.0.0 Risk Analysis", QSSI, 12/09/03.

Release 8.2.0.0
"RASS Release 8.2.0.0 Project Plan", QSSI, 10/09/03.
"RASS Release 8.2.0.0 Test Results and Evaluation Report", QSSI, 11/21/03.
"RASS Release 8.2.0.0 Functional Requirements Document", QSSI, 10/21/03.
"RASS Release 8.2.0.0 Program Specifications", QSSI, 11/14/03.
"RASS Release 8.2.0.0 System Specifications Document", QSSI, 10/21/03.
"RASS Release 8.2.0.0 Unit Test Plan", QSSI, 10/20/03.

Releases 8.0.0.0 & 8.1.0.0
"RASS Releases 8.0.0.0 & 8.1.0.0 Maintenance Manual", QSSI, 06/17/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Operations Manual", QSSI, 06/17/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Unit Test Plan", QSSI, 06/17/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 User's Manual", QSSI, 06/17/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 User's Manual Appendix", QSSI, 06/17/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 System Specifications Document", QSSI, 04/30/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Program Specifications Document", QSSI, 04/30/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Validation and Verification Plan", QSSI, 04/30/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Training Plan", QSSI, 04/30/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Database Specifications Document", QSSI, 04/30/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Data Requirements Document", QSSI, 03/18/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 Functional Requirements Document", QSSI, 03/18/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 System Support & Acquisition Plan", QSSI, 03/18/03.
"RASS Releases 8.0.0.0 & 8.1.0.0 System Security & Privacy Plan", QSSI, 03/18/03.
"RASS Release 8.0.0.0 Needs Statement", QSSI, 01/10/03.
"RASS Release 8.0.0.0 Project Plan", QSSI, 01/10/03.
"RASS Release 8.0.0.0 Feasibility Study", QSSI, 01/10/03.
"RASS Release 8.0.0.0 Cost/Benefit Analysis", QSSI, 01/10/03.
"RASS Release 8.0.0.0 System Decision Paper", QSSI, 01/10/03.
"RASS Release 8.0.0.0 Risk Analysis", QSSI, 01/10/03.

Release 7.1.0.0
"RASS Release 7.1.0.0 User's Manual", Accenture, 01/25/2002.
"RASS Release 7.1.0.0 Integration Test Results and Evaluation Report", Accenture, 01/25/2002.

“RASS Release 7.1.0.0 System Test Results and Evaluation Report”, Accenture, 01/25/2002.
“RASS Release 7.1.0.0 Release Notes”, Accenture, 01/22/2002.
"RASS Release 7.1.0.0 Test Plan", Accenture, 12/18/2001.
“RASS Release 7.1.0.0 Validation, Verification, and Testing Plan”, Accenture, 12/11/2001.
“RASS Release 7.1.0.0 System/Subsystem Specifications”, Accenture, 11/15/2001.
“RASS Release 7.1.0.0 Database Specifications”, Accenture, 11/15/2001.
“RASS Release 7.1.0.0 Program Specifications”, Accenture, 11/15/2001.
“RASS Release 7.1.0.0 System Support and Acquisition Plan", Accenture, 11/06/2001.
“RASS Release 7.1.0.0 Functional Requirements Document", Accenture, 11/06/2001.
“RASS Release 7.1.0.0 Data Requirements Document", Accenture, 11/06/2001.
“RASS Release 7.1.0.0 System Security and Privacy Plan", Accenture 11/06/2001.
“RASS Release 7.1.0.0 Needs Statement”, Accenture, 10/22/01.
“RASS Release 7.1.0.0 Project Plan”, Accenture, 10/22/01.
“RASS Release 7.1.0.0 Feasibility Study”, Accenture, 10/22/01.
“RASS Release 7.1.0.0 Cost/Benefit Analysis”, Accenture, 10/22/01.
“RASS Release 7.1.0.0 Risk Analysis”, Accenture, 10/22/01.

Release 7.0.0.0
"RASS Release 7.0.0.0 User's Manual", Accenture, 09/21/2001.
“RASS Release 7.0.0.0 System Decision Paper”, Accenture, 09/21/01.
"RASS Release 7.0.0.0 Maintenance Manual", Accenture, 09/21/2001.
"RASS Release 7.0.0.0 Operations Manual", Accenture, 09/21/2001.
"RASS Release 7.0.0.0 Installation and Conversion Plan", Accenture, 09/21/2001.
“RASS Release 7.0.0.0 Integration Test Results and Evaluation Report”, Accenture, 09/21/2001.
“RASS Release 7.0.0.0 System Test Results and Evaluation Report”, Accenture, 08/30/3001.
“RASS Release 7.0.0.0 Release Notes”, Accenture, 08/08/2001.
"RASS Release 7.0.0.0 Test Plan", Accenture, 07/19/2001.
“RASS Release 7.0.0.0 Training Plan”, Accenture, 06/20/2001.
“RASS Release 7.0.0.0 Validation, Verification, and Testing Plan”, Accenture, 06/20/2001.
“RASS Release 7.0.0.0 System/Subsystem Specifications”, Accenture, 06/08/2001.
“RASS Release 7.0.0.0 Database Specifications”, Accenture, 06/08/2001.
“RASS Release 7.0.0.0 Program Specifications”, Accenture, 06/08/2001.
"RASS Release 7.0.0.0 Project Schedule/ Workplan", Accenture, 05/31/2001.
“Accenture Quality Control Plan", Accenture, 05/31/2001.
“RASS Release 7.0.0.0 System Support and Acquisition Plan", Accenture, 05/14/2001.
“RASS Release 7.0.0.0 Functional Requirements Document", Accenture, 05/14/2001.
“RASS Release 7.0.0.0 Data Requirements Document", Accenture, 05/14/2001.
“RASS Release 7.0.0.0 System Security and Privacy Plan", Accenture 05/14/2001.
"RASS Release 7 Multifamily Functionality Business Requirements Document (FINAL)", Version 1.1, KPMG, 04/25/2001.
"FY2001 Resident Assessment Risk Assessment (Update)", Accenture, 05/15/2001.
"FY2001 Resident Assessment Feasibility Study (Update)", Accenture, 05/15/2001.
"FY2001 Resident Assessment Cost/Benefit Analysis (Update)", Accenture, 05/15/2001.
"FY2001 Resident Assessment Project Plan (Update)", Accenture, 05/15/2001.
"FY2001 Resident Assessment Needs Statement (Update)", Accenture, 05/15/2001.

1.5 Terms and Abbreviations

The following table defines terms and acronyms used throughout RASS SDM Define Phase Documentation.

Term	Definition
APP	Annual Performance Plan
BAH	Booz-Allen & Hamilton
BOP	Business Operating Plan
BRD	Business Requirements Document
DCG	Development Coordination Group
FRD	Functional Requirements Document
FYE	Fiscal Year End
HA	Housing Agency/Housing Authority
HUD	Department of Housing and Urban Development
HUDCAPS	HUD Central Accounting Processing System
HUDWeb	HUD's Intranet Web Site
JAD	Joint Application Development
MF	Multifamily
MFH	Multifamily Housing
NASS	Integrated Assessment Subsystem
OMB	Office of Management and Budget
PDF	Portable Document Format
PHA	Public Housing Agency/Public Housing Authority
PHAS	Public Housing Assessment System
PIC	PIH Information Center
PIH	Public and Indian Housing
POC	Point of Organizational Contact
QSSI	Quality Software Services, Inc.
RASS	Resident Assessment Subsystem
REAC	Real Estate Assessment Center
REACS	Real Estate Assessment Center System
SDM	Housing and Urban Development System Development Methodology
SOA	Section of the Act
TAC	Technical Assistance Center (formerly the Customer Service Center)
TARC	Troubled Agency Recovery Center
TRACS	Tenant Rental Assistance Certification System
TBD	To Be Defined
WASS	Web Access Security System

1.6 Points of Contact

1.6.1 Information

The following table lists Points of Organizational Contact (POCs) that may be beneficial for future reference.

Contact Name	Organization	Telephone Number
Lynda L. Zelnick	HUD – Contracting Officer	202-708-1772 ext.7126
Marian Louden	HUD – GTR	202-708-1817 ext.7165
Delton Nichols	REAC – RASS Project Manager	202-708-4932 ext.3115
Yangja K. Lee	REAC – RASS GTM/IT Manager	202-708-4932 ext.3081
Kevin N. Jones	REAC – RASS Assistant IT Manager	202-708-4932 ext.3111
Patrick Evans	BAH	202-708-4932 ext.3021
Gautam Ijoor	QSSI	202-708-4932 ext.3954
Alex Rozental	QSSI	202-708-4932 ext.3442
Eugene Lubarsky	QSSI	202-708-4932 ext.3152
Tanuj Sinha	QSSI	202-708-4932 ext.3332
Robin Hilton	QSSI	202-708-4932 ext.3453
Theresa Han	QSSI	202-708-4932 ext.3476
Robert Armstrong	QSSI	202-708-4932 ext.3458

Contact	Contact Information
RASS System Administrator	Delton Nichols, in coordination with the WASS subsystem, will facilitate RASS system administration as needed. 202-708-4932 ext.3115
REAC Technical Assistance Center	1-888-245-4860
REAC RASS E-mail address	REAC_RASS@hud.gov
REAC Internet Site	http://www.hud.gov/offices/reac/index.cfm
REAC RASS Internet Site	http://www.hud.gov/offices/reac/products/prodrass.cfm
REAC Intranet Site	http://hudweb.hud.gov/po/reac
HUD Customer Service Center Branch	1-202-708-3300

1.6.2 Coordination

Coordination must occur among the following organizations to successfully implement RASS Release 8.4.0.0:

Organization	Support Function
Paradigm Soln./BAH	Development Coordination, Installation, Deployment
HUD IT	Implementation Coordination, Installation, Deployment
Mandaree/Pearson	3rd Party Contractor: Resident Communication/Survey Support
QSSI	Requirements, Design, Development, Testing, Maintenance, Technical Support /Operations, Project Management
REAC	Business Requirements Support, Project Management
WASS	Security

Listed below are the coordination dates to which each Public Housing Assessment System (PHAS) must adhere in order to achieve the December 17, 2004 release date for Release 8.4.0.0.

Code Locked Date	Integration Test Start Date	Integration Test End Date	Final HARTS Request Submission Date	Release Date
10/19/2004	10/27/2004	11/30/2004	12/01/2004	12/17/2004

2.0 TEST EVALUATION

2 TEST EVALUATION

The test evaluation section covers RASS Release 8.4.0.0 functional requirements and String, System, Integration, and User Acceptance Test (UAT) performance criteria.

2.1 Requirements Traceability Matrix

A Requirements Traceability Matrix, included in Appendix A of this document, traces String, System, Integration, and User Acceptance Testing phase(s) to a system function, system requirement, test phase, and business rule.

RASS Release 8.4.0.0 Testing Conditions have been created based upon the Requirements Traceability Matrix. Please refer to Section 4.1 Testing Conditions for more information on Testing Conditions.

2.2 Test Evaluation Criteria

Test evaluation criteria describe the specific performance requirements that the String, System and Integration Tests must validate, including data processing and report generation. As previously mentioned, String, System and Integration Test Conditions have been created to test these performance criteria.

The following tables describe the RASS accuracy and timing features that meet the performance requirements imposed on the system.

PReq #	Performance Requirement	Source
1.0	**Accuracy**	
P1	RASS maintains automatic edits/validation of external data for approximately 3,200 PHA(s) and approximately 2,000 Multifamily properties	User Requirement

In order to test this performance requirement, testers will attempt to enter invalid data into RASS to ensure that automatic edits/validation of external data are performed correctly.

Preq #	Performance Requirement	Source
2.0	**Timing**	
P2	RASS will sample PHA unit addresses in a weekend batch process (presuming the batch process has been scheduled). If scheduled, this process will occur at 1:30am on Saturday and Sunday morning.	User Requirement
P3	RASS will score and flag PHA and sample and score MF using an NT script, which is kicked off on Saturday and Sunday at 12:00pm. Jobs will run sequentially beginning with PHA scoring, MF sampling and MF scoring if jobs have been scheduled for processing.	User Requirement
P4	RASS will be available for PHA component updates and certification 24 hours a day (based upon PHA access.)	User Requirement
P5	RASS will provide capability to generate multiple reports 24 hours per day.	User Requirement

P6	RASS will provide capability to load every page in 8 seconds or less (except for those screens that have approved waivers).	User Requirement
P7	RASS will provide capability to execute every stored procedure in 3 seconds or less (except for those with approved waivers).	User Requirement

The following is a list of the new and enhanced functionalities for RASS Release 8.4.0.0 that will be tested:

RASS Release 8.4.0.0 modifications will provide the capability for internal users to view a report containing PHAs that are at risk of failing RASS by several criteria. This report can be viewed by specific field office.

RASS Release 8.4.0.0 introduces the capability to address scoring and reporting changes due to PHAS regulation updates: Small PHA Deregulation, Consortia, FYE changes, etc.

RASS Release 8.4.0.0 will enhance the scoring process to provide the capability to automatically assign last assessment's scores for selected PHAs.

RASS Release 8.4.0.0 modifications will provide the capability for internal users to view Demographic report on Survey Section level.

RASS Release 8.4.0.0 will enhance PHA History Report to include the unit count. Also PHA History Report will specify when certification of Follow-up Plan not required.

RASS Release 8.4.0.0 will also enhance the edit functionality of the Implementation Plan to provide the capability to create a user specified window to restrict Implementation Plan Activities and verify the Implementation Plan Activities Date against the specified window.

RASS Release 8.4.0.0 will provide automatic e-mail notification to Field Offices and PHA when they fail to certify required RASS components.

For a detailed list of all the new and modified components and procedures subject to change for RASS Release 8.4.0.0, please refer to Section 2.1 System/Subsystem Description of RASS Release 8.4.0.0 System/Subsystem Specifications document.

The String, System and Integration Test environments will be set up in two separate testing regions that will each be used for their respective testing phases. String Test will have its own region; System and Integration Test will share the other region. All the testing environments will have a full cut of the production PHA data. The Integration testing region will be used to test a realistic load of PHA unit addresses, response rates, section scores, and question scores for the RASS sampling and scoring processes.

Test evaluation criteria will be met once all conditions outlined in the Test Conditions and Expected Results document (in Appendix B) have been successfully tested.

2.3 User System Acceptance Criteria

For the system to be considered fit for use it must pass Unit Testing, String Testing, System Testing, Integration Testing and User Acceptance Testing successfully without errors. RASS Release 8.4.0.0 User Acceptance Criteria has been outlined in the RASS Release 8.4.0.0 Testing Conditions and Expected Results document found in Appendix B of this document. RASS Release 8.4.0.0 User Acceptance Test (UAT) Testing Conditions outline the specific criteria that must be met in order for the system to successfully complete UAT.

3.0 TESTING SCHEDULE

3 TESTING SCHEDULE

The test schedule section will cover the overall RASS Release 8.4.0.0 test schedule, security, milestone information, equipment requirements, software requirements, personnel requirements, deliverables, testing tools, and supplies.

3.1 Overall Test Schedule

The RAS Release 8.4.0.0 workplan in Appendix C reflects the time duration of String Test, System Test, Integration Test, and User Acceptance Test. The workplan includes tasks, duration, begin and end dates, and resource allocation for the following components of testing:

- Documentation review;
- Data Preparation;
- Test Execution;
- Output review;
- System Release.

See Appendix C for specific details regarding the overall test schedule.

Also, please reference Section 3.3.2 Equipment Requirements and Section 3.3.4 Personnel Requirements for information concerning personnel involved in the test effort.

3.2 Security

Once the initial String Test environment is created, the RASS Development Team will assist in the testing effort but will not have access to the String Test environment. Only members of the String Test Team will have access to the environment. Migration of system components from the String Test promotion group in the PVCS Configuration Management Tool to the String Test environment will be controlled and performed solely by the Test Team. This will contribute to a valid String Test by ensuring that no one outside the Test Team has access to the String Test environment. In addition, System Testers will have sole ownership of the String Test environment to ensure that new versions of code are not introduced during passes of test execution.

Each tester will need update privileges on the testing region of the Sybase database in order to be able to modify test data as needed. Any User Ids needed for String, System, Integration, and User Acceptance Tests will be set up by DCG by request of the test team. A full Web Access Security System (WASS) or secure connection String Test will not be conducted in RASS Release 8.4.0.0 String, System, Integration, or User Acceptance Tests. The WASS Team will be responsible for a full Web Access Security System (WASS) or secure connection String Test. The WASS Team will determine the necessity of this test and the schedule of this test.

The following table lists the types of user Ids that exist in RASS, and describes the possible roles that can be assigned to those user Ids, along with corresponding HEREMS action codes for each role. RASS application-level security is enforced by assigning all users (including the RASS testing team) appropriate user Ids with corresponding roles and action codes assigned to them.

User ID Type	Description	Role Code(s)	Action Code(s)
M user ID	External MF only	MOA	RMR
M user ID	External PHA only	PCR	CFP, CIP, CUA, RPM, RPR
M user ID	External MF and PHA	PCR	CFP, CIP, CUA, RPM, RPR, RMR
H user ID	Internal	RCA	SBC
H user ID	Internal	RRM	AIN, AIT, AMP, ARS, ASY, CRI, CSP, DFP, DIP, DUA, PTC, RFP, RIP, RRM, RRR, RSD, RSR
I user ID	Non-REAC affiliated	RTP	SSR
Guest	Field Office Internal User	GUS	RIN, RSY, RMP, RSP, RRS, RTC, RRR, RRM, RFP, RIP

3.3 REAC Testing Location

This section provides a description of the testing locations.

Test to be Performed	Test Location	Participating Organizations
String Test	HUD REAC, Portal Building 1250 Maryland Ave, SW Washington, DC 20024	QSSI Requirements and Development Teams; Development Coordination Group (DCG) Team
System Test	HUD REAC, Portals Building 1250 Maryland Ave, SW Washington, DC 20024	QSSI Requirements and Development Teams; Development Coordination Group (DCG) Team
Integration Test	HUD REAC, Portals Building 1250 Maryland Ave, SW Washington, DC 20024	QSSI Requirements and Development Teams; Development Coordination Group (DCG) Team
User Acceptance Test	HUD REAC, Portal Building 1250 Maryland Ave, SW Washington, DC 20024	QSSI Requirements and Development Teams; RASS Project Management

3.3.1 Milestone Chart

The following table depicts the activities and events to be conducted for String, System, Integration, and User Acceptance Tests. For the complete RASS Testing Workplan, refer to Appendix C Workplan.

Activities and Events	Start Date	End Date
Validation, Verification, and Testing Plan	07/20/2004	07/28/2004
Unit Test*	07/10/2004	09/10/2004
Test Prep Activities (String, System, Integration, User Acceptance)	07/15/2004	09/06/2004
String Test	09/20/2004	10/11/2004
System Test	10/12/2004	10/26/2004
Integration Test	10/27/2004	11/25/2004
User Acceptance Test	11/30/2004	12/14/2004
Test Results Document (System Test, Integration pass in integrated environment and UAT)	11/25/2004	12/08/2004

**Please Note: RASS Release 8.4.0.0 Unit (Component) Test information will be provided in RASS Release 8.4.0.0 Unit Testing Plan.*

3.3.2 Equipment Requirements

Testers will be using HUD provided hardware and software to complete RASS Release 8.4.0.0 System and Integration Tests. RASS internal users will perform RASS Release 8.4.0.0 User Acceptance Test using HUD provided hardware and software. One UAT Coordinator will oversee the User Acceptance Test.

Please reference the following two tables for specific details:

System Test Environment (Release 8.4.0.0: 10/12/2004 – 12/14/2004):

Hardware	www6-reac-syst
OS	Win2K
Application server	Allaire's ColdFusion 5.0
Webserver	IPlanet Web server 7.0
LDAP	Netscape Directory Server 5.2
Machine Type	Dell 6450
CPU	4x700 MHz
RAM	4GB RAM
HDD	4x36GB

Integrated Test Environment (Release 8.4.0.0: 10/21/2004 – 11/30/2004):

Hardware	www6
OS	Win2K
Application server	Allaire's ColdFusion 5.0
Webserver	IPlanet Web server 7.0
LDAP	Netscape Directory Server 5.2
Machine Type	Dell 6450
CPU	4x700 MHz
RAM	3GB RAM

Hardware	www6
HDD	4x36GB

3.3.3 Software Requirements

Other than the testing template, the PVCS Configuration Management tool, Workstation-FTP, the SQL Advantage tool, and the PVCS Quality Control Request Tracking Database, additional software is not required to support the String, System, Integration and User Acceptance testing. Please refer to Section 3.3.6 Testing Tools for more information on the software used as testing tools.

3.3.4 Personnel Requirements

The following table provides a listing of the personnel necessary to complete the RASS Release 8.4.0.0 String, System, Integration and User Acceptance testing effort. This table includes personnel that will perform testing and manage the testing effort for RASS Release 8.4.0.0.

Name	Title	Testing Experience	Location
Tanuj Sinha	Analyst	10 months	HUD-Portals, Washington, DC
Alex Rozental	Analyst	6 months	HUD-Portals, Washington, DC
Theresa Han	Tester	10 months	HUD-Portals, Washington, DC
Robin Hilton	Tester	10 months	HUD-Portals, Washington, DC

3.3.5 Deliverable Materials

The following RASS Release 8.4.0.0 String, System, Integration, and User Acceptance Test deliverables will be provided to the client for sign-off:

- HUD SDM Design Phase: Validation, Verification, and Testing Plan (includes Conditions and Expected Results)
- HUD SDM Evaluate Phase: Test Results and Evaluation Report

3.3.6 Testing Tools

This section identifies the testing tools to be used in the preparation and execution of the String, System, Integration and User Acceptance Tests. The REAC currently owns licensed versions of these testing tools.

- *Testing Templates* are used to create test conditions, test cycles, and test scripts.

- *PVCS Version Manager* is a configuration management tool. The system components (i.e. source code, reference tables, security protocols, etc.) for RASS Release 8.4.0.0 are stored in PVCS Version Manager. The system components required for String, System and Integration Tests will be migrated from the Unit Test environment to all subsequent testing environments using the PVCS tool.

- *Workstation – FTP* is used to transfer files. The system components required for String, System and Integration Tests will be migrated from the Unit Test environment to all subsequent testing environments. Internet Explorer and Netscape Browsers will be used to test RASS Release 8.4.0.0 functionality.

- *SQL Advantage* is used to query the database. Pre-defined queries will be executed to confirm that actions performed on data are made to the proper tables and to the proper data.

- *PVCS Tracker Quality Control Request Tracking Database* is the problem report tracking tool. Any software deficiency found and verified will be entered into the PVCS Quality Control Request Tracking database, which will allow for full tracking from creation through resolution.

3.3.7 Site Supplied Materials

HUD REAC supplies the test team with all the software, hardware and office supplies needed to complete RASS Release 8.4.0.0 String, System, Integration and User Acceptance Tests. The team will be using basic work supplies (PC, paper, etc.) and HUD REAC supplies (SQL Advantage, PVCS, etc.). Please refer to Section 3.3.6 Testing Tools for more information on the software that will be required.

4.0 TESTING CHARACTERISTICS

4 TESTING CHARACTERISTICS

This section will cover the extent of testing to be conducted, test constraints, test progression, and test evaluation.

4.1 Testing Conditions

The RASS Release 8.4.0.0 String, System, Integration, and User Acceptance Test (UAT) Testing Conditions and Expected Results document, provided in Appendix B of this document, covers functionality from RASS Releases 1.0.0.0 through 8.4.0.0. The Testing Conditions have been created using the Requirements Traceability Matrix, found in Appendix A of this document. In the RASS Release 8.4.0.0 Test Conditions and Expected Results document, the system functions and system requirements (from the Requirements Traceability Matrix) are mapped to the conditions that will be tested in String, System, Integration and User Acceptance Tests. High-level system functions have been included in the Test Conditions and Expected Results document in order to trace clearly back to the system functions found in the Requirements Traceability Matrix. For more information on the Requirements Traceability Matrix, please refer to Appendix A Requirements Traceability Matrix.

RASS test efforts will be made using normal database input and, when appropriate, special input for error and exception processing.

RASS Release 8.4.0.0 String, System, Integration, and User Acceptance Tests will be conducted using predetermined input parameters and database query procedures as outlined in Section 5.1 String Test.

4.2 Extent of Testing

RASS Release 8.4.0.0 String, System, Integration, and User Acceptance Tests will be considered successful when the actual test results match the expected results outlined in the Test Conditions and Expected Results document (Appendix B). Deviations between the actual results and expected results are documented in PVCS Quality Control Request Tracking Database. These deviations will be documented as a System Change Request (SCR) ticket, which will outline the steps taken to create the deviation and trace the deviation back to the test condition where it was originally created. Please refer to Section 5.1.5 Test Procedures for more information on the procedures involved in documenting issues and problems in the PVCS Tracker. If the problem cannot be resolved or recreated, the Requirements Team Lead and Project management will review the deviation. Project management will make the final decision on the resolution of the deviation.

For each pass where actual results do not match expected results and a code fix is required, a spot check of the code fix will be performed to ensure that the problem was resolved and that no other functionality was impacted as a result of the code fix. Once the spot check is complete, a regression pass will be completed. The regression pass will ensure no subsequent errors were made in the effort to fix the original discrepancy.

Additionally, all requirements from previous RASS releases will be Integration tested. However, the Integration tests will not consist of full vertical tests, but high-level horizontal tests to verify no requirements have been effected by new and/or modified functionality. Testing will continue until no

deviations or problems are found and a clean pass of testing is achieved. Please refer to Section 5.1 String Test for more information.

Normal database input and, in some cases, special input for error and exception processing will be used during RASS testing efforts.

4.3 Data Recording

In order to fulfill the data-recording requirement, screen prints will be taken of the actual test results in the final, clean pass of String and Integration Tests. These screen prints will correspond to steps in the System and Integration Test scripts.

DCG will create a full production cut of RASS data to populate the String Test environment that may be referenced in subsequent passes. DCG will perform a refresh of the testing database for each pass of testing. Once the test environment is verified as accurate, DCG will perform a backup to ensure the environment can be refreshed as quickly as needed.

4.4 Testing Constraints

Testing Scheduled Time Frames	Start Date	End Date
String Test in the RASS environment	09/20/2004	10/11/2004
System Test	10/12/2004	10/26/2004
Integration Test	10/27/2004	11/25/2004
User Acceptance Test	11/30/2004	12/14/2004

Any extensions to these time frames will directly correlate to the timely completion of the software components. It is expected that most issues will be uncovered in the first passes of the String Test and that the later passes will encounter significantly fewer issues. The testing window, nonetheless, does not allow much time for delays in execution. Any delays in the HUD server, delivery of the software, creation of the test environment/region, fixes to identified software problems or completion of the String, System or Integration Tests may impact the test schedule.

4.5 Test Progression

Test progression will ensure that the cycles for each test are completely and successfully performed prior to beginning the next test. This test progression improves the possibility that issues or problems are resolved early in the test schedule. The String Test will not be completed or signed off upon until a clean pass of String Testing has been achieved. A clean pass is executing all the scripts without errors and logging no System Change Requests (SCRs). Each phase of testing will continue until a clean pass is achieved. Section 5.1 String Test explains the progression made from String Test to Integration Test. Similarly, Integration Test will not be completed until a clean pass of Integration Testing has been achieved.

4.6 Test Evaluation

This section covers the test data criteria, deviation tolerance, number of test interruptions allowed and methods to reduce test data.

4.6.1 Test Data Criteria

RASS Release 8.4.0.0 String and Integration Test results will be evaluated by comparing the actual test results to the expected results outlined in the Test Conditions and Expected Results document (Appendix B). Once all the actual results are equal to the expected results, the test will be considered successful. For RASS Release 8.4.0.0 User Acceptance Test, the business users will record their comments, which will be considered the UAT test results. RASS Project Management will evaluate these comments as the data criteria for User Acceptance Test.

4.6.1.1 Tolerance

The RASS Test Team will not tolerate deviations where actual test results do not match the expected results outlined in the Test Conditions and Expected Results document in Appendix B. In addition, the RASS Test Team anticipates zero tolerance for deviations from RASS Release 8.4.0.0 scope. However, if a question arises concerning a deviation from the scope, RASS Project Management will evaluate the change request. This evaluation will be performed on a case-by-case basis, and may vary depending upon the identified issue.

For reports that do not meet REAC performance standards, a waiver form approval is required. REAC performance standards state that pages must load in 8 seconds or fewer and stored procedures must execute in 3 seconds or fewer. The waiver must be filled out and signed by the appropriate REAC employees for the page to be considered waived from the time constraint requirements.

4.6.1.2 System Breaks

Interruptions to String, System, Integration and User Acceptance Tests will only be allowed for server downtime and DCG database maintenance. The maximum overall time delay during the testing schedule totals a one-day delay (one hour for the UAT) due to interruptions or system breaks. Any delay greater than one day (one hour for the UAT) could cause delays in the testing schedule.

4.6.2 Test Data Reduction

The String, System, and Integration Test data set up in the integrated environment on server WWW6 will contain a full cut of production data that includes all PHAs. This production data set will be in a form suitable for evaluation and testing. The screen prints taken in the final, successful pass of the String and Integration Tests will be used to verify test results.

Data will be evaluated and documented in test scripts. The data elements will be manipulated during the writing of the test scripts so as to be suitable for use during the testing cycles.

5.0 TEST DESCRIPTION

5 TEST DESCRIPTION

This section provides a description of the String, Integration, and User Acceptance (UAT) Tests.

5.1 String Test

This section provides a description of the String Test.

The objectives of RASS Release 8.4.0.0 String Test include performing a detailed test of the new capabilities implemented in RASS Release 8.4.0.0 as well as a high-level test of all RASS functional Public Housing requirements. At the conclusion of this String Test, the system should support all functionality required to generate a RASS score without error, including the new functionality of RASS Release 8.4.0.0. String Test will be conducted by RASS testers using detailed test scripts, focusing on the new capabilities for RASS Release 8.4.0.0.

To view the Release 8.4.0.0 functionality that will be tested in the String Testing efforts, reference Section 1.2 Scope of this document.

The String Test is a critical step towards full deployment of RASS Release 8.4.0.0. With String Test scheduled to be completed eight weeks prior to deployment, the String Test approach is designed to streamline the discovery and resolution of any software deficiencies. As a result, the following criteria are essential for a successful test:

1. Rapid discovery of critical software deficiencies;
2. Complete validation of software component integration throughout the business processes;
3. Complete systematic testing for software deficiencies;
4. Timely entries and updates (including issue closure) to the new PVCS Tracker Quality Control Database by testers and developers, respectively, to ensure thorough tracking and documentation of all deficiencies from discovery through resolution.

RASS Release 8.4.0.0 String Test will utilize a three-level, three-pass approach. Each level of testing includes one or more cycles. A cycle is a group of related test conditions that support a RASS business process. An example of a cycle would be the internal and external RASS Survey Administrator processes. The cycle would include all the test conditions associated with generating a RASS score at the Development-level.

Each cycle within each level of testing will be run at least three times (i.e., three passes). The objective of pass 1 is to get through the test as quickly as possible, finding as many defects as possible and implementing code fixes where needed. The objective of pass 2 is to integration test the defects fixed from pass 1 and determine if the pass 1 code fixes caused any more defects. The third pass should run without any defects. The three-pass approach is not absolute – passes will continue to be executed until a cycle runs without any defects.

This approach focuses on finding and identifying as many problems as possible, as early as possible so that there will be adequate time to fix and retest them. This approach also makes it easier to predict when testing will be complete and ensures that all defects are completely fixed and these implemented

fixes do not result in other errors. The Development/Unit Test environment must also be maintained in order to integration test defects found in the String Test. For each code fix, a complete Unit Test will need to be re-executed in the development environment. Any new conditions created, as a result of fix implementations, will need to be added to the existing set of Test Conditions.

The three-level, three-pass technique is a structured approach, which:

1. **Facilitates predicting when testing will be complete.** This approach focuses on finding as many problems as possible as early as possible, so there will be time to fix and retest them. Early in the process, the team will get a sense of what to expect throughout the rest of the test; this in turn allows the team to predict when testing will be complete.

2. **Reduces overall elapsed time to test.** This approach recommends a mix of independent and dependent cycles. A small number of test cycles should be dependent in order to ensure that data can pass through the entire application correctly. The remainder of the test cycles should be independent so that a single critical defect does not halt application product test altogether. This will reduce the overall time frame to test.

3. **Ensures thorough/quality testing.** The three-pass component of this approach builds in integration test runs to ensure that defects are completely fixed and that the fixes did not break anything else.

This String Test will apply the three-level, three-pass approach as described in the next four sections. Additional passes in the String Testing environment will be added only if these passes are necessary in order to achieve a clean pass.

The String Test will focus on both screen (front-end) and database (back-end) functionality. Validation of business rules and logic will help ensure that the String is functioning as designed. A successful end-to-end flow will be measured by comparing expected results to actual results. This approach ensures the following: 1) the software sufficiently supports the user in successfully completing RASS internal and external processes and 2) REAC captures the appropriate data successfully.

5.1.1 System Functions

The Requirements Traceability Matrix in Appendix A details the system functions that will be tested. The Test Conditions and Expected Results document in Appendix B traces these system functions to the conditions that will be tested.

5.1.1.1 Level 1

Level 1 will be a high-level horizontal String Test of all the Public Housing functional areas in RASS. The purpose of this high-level test is to test data flow throughout the entire application. This test ensures that the environment and code are stable before starting the Level 2 String Test execution. This test also ensures that all Test Conditions, provided in Appendix B of this document, have been tested successfully.

The Level 1 Cycles are defined in the table below:

Level 1: High-Level Horizontal	
Cycle 1	Internal and External RASS Public Housing Processes
Cycle 2	Internal and External Survey Administrator Processes

For example, in Cycle 1: Internal and External RASS Public Housing Processes, one PHA will be taken through the high-level functions necessary to generate a RASS score. These high-level processes include unit address sampling, survey scoring, reporting results review, and RASS score approval.

5.1.1.2 Level 2

Level 2 String Test will be a series of detailed vertical tests of RASS Release 8.4.0.0 functional areas. Testing the Field Office Dynamic Report Selection screen is an example of Level 2 testing. These vertical tests can be independent tests, which allow for concurrent testing, and thereby reduce the overall testing time. Level 2 testing will be started only once Level 1 testing has determined that the environment and code are stable.

Level 2 will consist of the following cycles:

Level 2: Detailed Vertical	
Cycle 1	Internal and External RASS Public Housing Processes
Cycle 2	Access At-risk Report
	Access PHA Report by flagged categories
	Access PHA History Report
	Access the Demographic Report
	Assign last assessment's score
	Assign Implementation Activities dates

Level 2 testing will focus on a detailed test of new capabilities implemented in RASS Release 8.4.0.0, as outlined in the above table.

At the beginning of each pass of Level 1 and Level 2 cycles, the String Test database will be refreshed with a copy of a clean data set. When the pass, including all Level 1 and Level 2 cycles, is completed, the String Test database will be refreshed to its original starting point. At this point, the String Test Team will enter a complete set of submission data to assure that any fixes completed during the prior pass did not affect any functionality in new pass.

5.1.1.3 Level 3

Level 3 String Test is a series of error processing and exception handling tests. Testing how the application handles the entry of incorrect user IDs and/or passwords is an example of an error-processing test.
The Level 3 cycles are defined in the table below:

Level 3: Error Processing/Exception Handling	
Cycle 1	Internal and External RASS Public Housing Error Processing and Exception Handling
Cycle 2	Internal and External Survey Administrator Error Processing and Exception Handling

5.1.2 Test/Function Relationships

RASS Release 8.4.0.0 String Test Levels that will constitute the overall test activity are described in Section 5.1.1.1 Level 1 through Section 5.1.1.3 Level 3.

5.1.3 Means of Control

For RASS Release 8.4.0.0, testers will manually execute the String Test scripts.

5.1.4 Test Data

Data sets will be created for every scenario to be tested. Since much of RASS testing is scenario-based, one test script will support multiple data sets. In addition, multiple scripts may be supported by a single data set. Query access to the test databases will be required so that the String Test Team can confirm data modifications made to tables in the database using the SQL Advantage Tool. The String Test Team will query the database before and after each test script step that requires the user to add, update or delete data in the database. These queries will verify that the proper action was performed in the proper table on the proper data.

For String, System and Integration Testing passes in the integrated environment, the database on server WWW6 will be pre-populated with a full cut of production data for all PHAs. Please refer to Section 5.1.5.2 Setup for more information regarding the setup of the integrated environment for Integration Testing of RASS Release 8.4.0.0.

Although the data used for RASS is not highly sensitive data, because the testing environment contains a copy of production data and may contain real PHA scores, a need for security considerations exists. Please refer to Section 3.2 Security for a detailed discussion of the application and database security pertaining to Subsections 5.1.4.1-5.1.4.4.

5.1.4.1 Input Data

The input data will be included in the sample test data mentioned in Section 5.1.4 Test Data.

5.1.4.2 Input Commands

As REAC does not currently employ an automated testing tool, the initialization of tests, the interruption of tests, the repetition of unsuccessful or incomplete tests, or the termination of tests will be controlled manually by the RASS System Test Team.

5.1.4.3 Output Data

The output data will be maintained in the final, clean pass of the String Test using screen prints. These screen prints will be used to compare actual test results with expected test results. These screen prints will be delivered to the RASS HUD IT manager after the final, successful pass of the String Test. The data will be located in the database on the WWW6 server, d016.

5.1.4.4 Output Notification

For sampling, survey result upload, validation, and scoring, e-mail notification will be sent to the REAC RASS System Lotus Notes mailbox. There will be no automated output notification from RASS Release 8.4.0.0 System Test other than these e-mails. The System Change Requests (SCRs) documented in the PVCS Quality Control Database will indicate irregularities in system output.

5.1.5 Test Procedures

5.1.5.1 Procedures

This section describes step-by-step procedures followed by System Testers.

1. Each tester will be assigned a script to complete.
2. The tester follows the test script and executes the test.
3. When the expected results are equal to the actual results the tester must indicate that each step was completed successfully.
4. Each step in the test script will have an associated screen print. (Final pass only.)
5. If the actual result of the test does not match the expected result, the tester will create a SCR using the PVCS Tracking Quality Control Request Tracking database. The tester will attach screen prints of the issue or problem to this SCR, including database query results before and after the problem was encountered. The tester will monitor the progress of the SCR by referencing the SCR. The tester will follow the procedures diagramed below.
6. After a fix is implemented, the tester will verify that the problem has been resolved and no subsequent errors were made in the fix process.
7. When all the fixes for a cycle have been implemented, the tester will begin Integration testing. Integration testing ensures the fixes did not affect any other functionality of the system.
8. When all passes are complete, the tester will organize the screen prints.
9. The test script, test data, and screen prints will be saved for future reference.

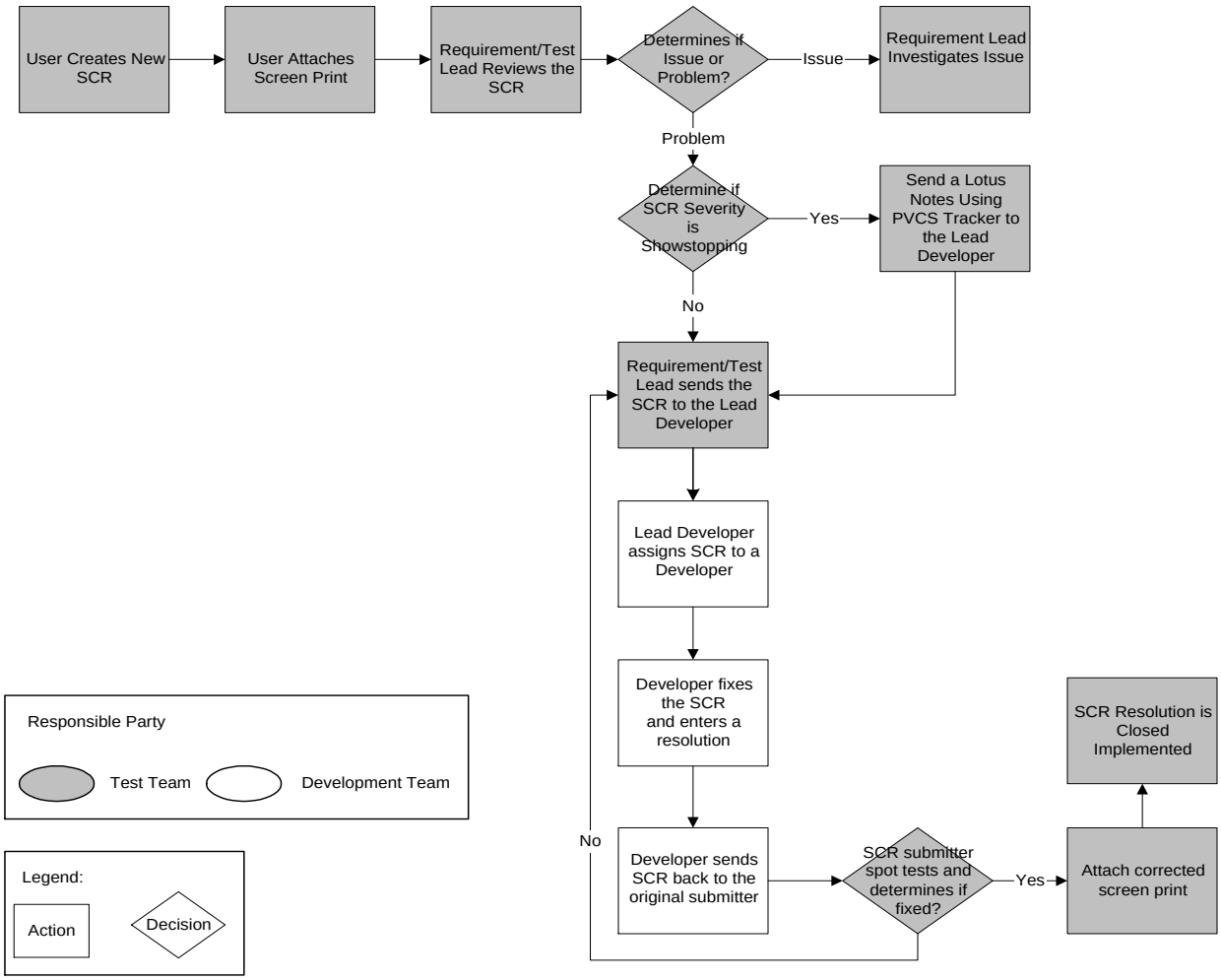
When a discrepancy is found between the actual and expected results in any of the tests, the deficiency will be tracked in the PVCS Quality Control Database.

Any software deficiency found and *verified* will have a System Change Request (SCR) created in the PVCS Quality Control Database, which will allow for full tracking from creation through resolution. Test Team members will designate these requests as either an Issue or a Problem. The test teams will be the only teams opening and closing SCRs during the String, System and Integration Tests. This contributes to clear communication between the testing and development teams. The test teams will only close SCRs that have successfully completed an integration test to verify the fix was successfully implemented.

Please refer to the diagram on the following page for more information on the procedures taken when a discrepancy is found.

5.0 Test Description

The following diagram shows the information flow and responsibilities for the database.



5.1.5.2 Setup

The String Test environment will be setup two weeks after the setup of the Development environment. The String Test environment includes a full cut of production data, the system components for RASS Release 8.4.0.0 have been migrated into the String Test environment from the Development/Unit test environments. To immediately verify the test environment, Testers should execute past horizontal scripts, including comparisons of the reference tables to production. Once the test environment is verified as accurate, DCG will create a backup of the test database to ensure that it can be refreshed quickly in the future as needed for this test.

The integrated environment in which all the testing is to be conducted will be setup by DCG on server WWW6 and will include a full cut of RASS production data. This production data set will be created in the integrated environment prior to the migration of the system components for RASS Release 8.4.0.0. To immediately verify the test environment, Testers should execute past horizontal scripts, including comparisons of the reference tables to production. Once the test environment is verified as accurate, DCG will create a backup to ensure that it can be refreshed quickly in the future as needed for this test.

All system components required for String Testing (i.e. source code, reference tables, security protocols, etc.) will be migrated from the Development/Unit Test environment to the applicable testing environments using the PVCS configuration management tool once Development/Unit Test is completed. An inventory of all required components will be compared to the testing environment once the migration has been completed to ensure that the environment contains all required but no extraneous components. Migration of system components from the Development environment to the String, System and Integration Test environments should follow the configuration management process. This will ensure that testing is being executed on the most up-to-date version of the software.

All system components required for System Test and Integration Test (i.e. source code, reference tables, security protocols, etc.) will be migrated from the String Test environment to the testing environment on server WWW6 before the system and integration pass in the integrated environment is started.

5.1.5.3 Initialization

String Test will be initialized when the test environment has been refreshed with a copy of the production data set and a copy of the new system source code components. Initialization indicates when a new pass of testing may commence.

DCG will initialize the test environment between passes by refreshing the String Test database upon the request of the Test team. The PVCS Version Manager will be used to migrate new system source code components to the String Test environment.

5.1.5.4 Preparation

The following will occur in preparation for RASS Release 8.4.0.0 String Test:

- The RASS Release 8.4.0.0 Validation, Verification, and Testing Plan (HUD SDM Design Phase), including the Test Conditions and Expected Results document will be reviewed and delivered.
- The String Test scripts and data will be created and reviewed.

- Testers will have a set of internal and external user ids.
- The String Test environment will have a full cut of production data with the system components migrated into the String Test environment from PVCS Version Manager.

5.1.5.5 Termination

RASS Release 8.4.0.0 String Test will end when all of the String Test scripts have been completed successfully, that is when the actual test results match the expected results, as outlined in the Test Conditions and Expected Results document. The actual test results and the screen prints of the final clean pass will be compiled and given to project management.

5.2 System and Integration Test

This section provides a description of System Test and Integration Test.

The objectives of RASS Release 8.4.0.0 System Test and Integration Test include a full test of the new capabilities implemented in RASS Release 8.4.0.0 as they affect other PHAS subsystems, in addition to testing any high-level touch points between RASS and NASS. At the conclusion of these tests, the system should support all functionality required across each PHAS Subsystem without error. RASS Release 8.4.0.0 System Test and Integration Test will be conducted by RASS testers using Integration Test scripts.

To view the new functionality and upgraded capabilities to be implemented as part of RASS Release 8.4.0.0 that will be tested in the System and Integration Testing efforts, reference section 1.2 Scope of this document.

In addition, the RASS System Test and Integration Test Team will work closely with the Development Coordination Group (DCG) to verify that the sampling and scoring programs have functioned correctly in the REAC Nightly batch.

With RASS Release 8.4.0.0 System Test and Integration Test beginning 6 weeks prior to deployment and only after a successful String Test, the testing approach is designed to streamline the discovery and resolution of integration deficiencies found in the functionality and logic of RASS Release 8.4.0.0.

In order to discover and resolve the system and integration software deficiencies, the System Test and Integration Test will follow an in-depth, two level, six-pass approach (two passes of System Test and four passes of Integration Test) where the level and pass definitions are identical to those outlined in the String Test Overview. It focuses 1) on entering valid data into RASS and producing and indicator score and 2) on validating RASS data in the Integrated Assessment Subsystem (NASS).

This System Test and Integration Test will apply the two-level, six-pass (two passes of System Test and four passes of Integration Test) approach as follows:

Level 1: High-Level Horizontal	
Cycle 1	Internal and External RASS processes in the integrated environment
Cycle 2	RASS data as recorded in NASS

For example, Cycle 1: Internal and External RASS processes in the integrated environment would test the generation of a overall RASS score for one PHA and one Multifamily Property in the System Test and Integration Test environment.

Level 2: Detailed Vertical	
Cycle 1	RASS/NASS Bridge Functionality

For example, Cycle 1: RASS/NASS Bridge Functionality would focus on the touch points between the RASS subsystem and NASS. The processing of an invalidated PHA indicator score is an example of a touch point that would be tested in detail during Level 2 System and Integration Testing.

5.2.1 System Functions

The RASS Release 8.4.0.0 Requirements Traceability Matrix in Appendix B details the system and integration functions that will be tested. The Test Conditions and Expected Results document in Appendix B trace these system functions to the conditions that will be tested.

5.2.2 Test/Function Relationships

RASS Release 8.4.0.0 System Test and Integration Test will constitute the overall test activity and are described in Section 5.2 System and Integration Test.

5.2.3 Means of Control

RASS Release 8.4.0.0 System Test and Integration Test activities will be controlled manually by the RASS Integration Tester.

5.2.4 Test Data

RASS Release 8.4.0.0 System and Integration Test data will be prepared prior to the test execution, similar to the test data prepared for the String Test. Please refer to Section 5.1.4 Test Data for more information.

In addition, PHAs will be selected for use by all the PHAS subsystems (RASS, MASS, PASS and NASS) during System and Integration Testing. These PHAs will be used by the PHAS subsystems to generate PHAS scores as part of the System Test and Integration Test. Additional PHAs may be used by RASS as necessary, but each PHAS subsystem must provide data for the selected PHAs to ensure a PHAS score is generated during the Integration Test effort.

5.2.4.1 Input Data

The input data will be included in the sample test data mentioned in Section 5.2.4 Test Data.

5.2.4.2 Input Commands

As REAC does not currently employ an automated testing tool, the initialization of tests, the interruption of tests; the repetition of unsuccessful or incomplete tests, and the termination of tests will be controlled manually by the RASS System and Integration tester.

5.2.4.3 Output Data

The output data will be maintained in the final clean pass of the Integration Test using screen prints. These screen prints will be used to compare actual test results with expected test results. These screen prints will be delivered to the RASS HUD IT manager after the final, successful pass of the Integration Test. The data will be stored in the database on the PO8A server.

5.2.4.4 Output Notification

There will be no automated output notification from RASS Release 8.4.0.0 System Test and Integration Test. The System Change Requests (SCRs) documented in the PVCS Quality Control Database will indicate irregularities in system output.

5.2.5 Test Procedures

5.2.5.1 Procedures

The procedures followed by RASS testers during System Test and Integration Test closely match the procedures followed during the String Test. The issue tracking procedures followed by RASS testers during the System Test and Integration Test will also closely match the procedures followed during the String Test. Please refer to Section 5.1.5.1 Procedures for more information.

5.2.5.2 Setup

The System Test and Integration Test environment will be setup by DCG and will include a full cut of RASS production data. This production data set will be created in the System Test and Integration Test environment prior to the migration of the system components for RASS Release 8.4.0.0. To immediately verify the test environment, testers should execute past horizontal scripts, including comparisons of the reference tables to production. Once the test environment is verified as accurate, DCG will create a backup to ensure that it can be refreshed quickly in the future as needed for this test.

All system components required for the System Test and Integration Test (i.e. source code, reference tables, security protocols, etc.) will be migrated from the String Test environment to the applicable testing environment once Development/Unit Test is completed.

5.2.5.3 Initialization

System Test and Integration Test will be initialized when the test environment has been refreshed with a copy of the production data set and a copy of the new system source code components. Initialization indicates when a new pass of testing may commence.

DCG will initialize the test environment between passes by refreshing the System Test and Integration Test database at the conclusion of each pass. The PVCS Version Manager will be used to migrate new system source code components to the System Test and Integration Test environments.

5.2.5.4 Preparation

The following will occur in preparation for RASS Release 8.4.0.0 System Test and Integration Test.

- The RASS Release 8.4.0.0 Validation, Verification, and Testing Plan (HUD SDM Design Phase), including the Test Conditions and Expected Results document will be reviewed and delivered.
- The Integration Test scripts and data will be created and reviewed.
- Testers will have a set of internal and external user ids.

- String Tests will be complete.
- The System Test and Integration Test environment will have a copy of production with the system components migrated into the System Test and Integration Test environments from PVCS Version Manager.

5.2.5.5 Termination

RASS Release 8.4.0.0 System Test and Integration Test will end when all the Integration Test scripts have been completed successfully, and the actual test results match the expected results outlined in the Test Conditions and Expected Results document (in Appendix B). The actual test results and the screen prints of the final clean pass will be compiled and given to project management for signoff.

5.3 User Acceptance Test

This section provides a description of the User Acceptance Test (UAT).

The objectives of RASS Release 8.4.0.0 UAT include conducting a review of revised internal RASS applications, receiving RASS REAC management feedback on the revised applications, identifying areas for improvement for future RASS application releases, and answering any RASS application questions.

To view the new functionality and upgraded capabilities to be implemented as part of RASS Release 8.4.0.0 that will be tested in the User Acceptance Testing efforts reference Section 1.2 Scope of this document.

RASS Release 8.4.0.0 UAT will take approximately a week for preparation and one day for execution. RASS UAT is currently scheduled to take place a short time before the deployment for each release for the RASS business team; however, if possible, this testing will be moved up to an earlier date.

During the UAT, RASS Requirements Test Team will provide the UAT participants (RASS Project Team members) with an overview of the UAT and with detailed instructions in order to execute the UAT. In addition, it will be explained to UAT participants that their feedback on RASS's navigation, functionality, and usability is essential.

The UAT will begin once all of the users' preliminary questions or concerns have been acknowledged. Internal UAT participants will be given access to the testing environment in RASS and given time to use the system and test the system functionality. Comments will be solicited from all users and documented after the test. In addition, the RASS Requirements Test Team will note any of the users' verbal comments concerning usability issues and/or enhancement suggestions. The RASS Requirements Test Team will also be available to offer suggestions on how to use the system efficiently.

5.3.1 System Functions

The RASS Release 8.4.0.0 Requirements Traceability Matrix in Appendix B details the UAT functions that will be tested. The Test Conditions and Expected Results document in Appendix B trace these system functions to the conditions that will be tested.

5.3.2 Test/Function Relationships

The UAT Tests that will constitute the overall UAT activity are described in Section 5.3 User Acceptance Test.

5.3.3 Means of Control

The User Acceptance Test activities will be controlled manually by RASS UAT participants (RASS Project Team members).

5.3.4 Test Data

For the internal UAT, RASS Release 8.4.0.0 UAT test data will be created prior to the test execution. Data sets will be created for every scenario to be tested. Since much of RASS testing is scenario-based, one test script will support multiple data sets. In addition, multiple scripts may be supported by a single data set.

5.3.4.1 Input Data

The input data will be included in the sample test data mentioned in Section 5.3.4 Test Data.

5.3.4.2 Input Commands

As REAC does not (at this point) employ an automated testing tool, the initialization of tests, the interruption of tests, the repetition of unsuccessful or incomplete tests, and the termination of tests will be controlled manually by RASS UAT participant and the RASS Test Team member assisting with UAT.

5.3.4.3 Output Data

The output data will be analyzed in the UAT using the feedback and comments of the UAT participants. The feedback of UAT participants will be used to compare actual test results with expected test results.

5.3.4.4 Output Notification

There will be no automated output notification from RASS Release 8.4.0.0 User Acceptance Test. The System Change Requests (SCRs) documented in the PVCS Quality Control Database will indicate irregularities in system output.

5.3.5 Test Procedures

5.3.5.1 Procedures

This section describes step-by-step the UAT procedures performed by user acceptance participants and RASS testers.

1. Each UAT participant will be assigned a list of steps to follow, through the use of prepared scripts.
2. Each UAT participant will walk through each script.
3. The UAT participant will document any issues, problems or recommendations for future RASS releases.
4. The UAT instructions and UAT participant comments will be saved for future reference.

During RASS UAT, issues will be documented by the UAT testing participants and RASS team members. At the conclusion of the UAT, the RASS Requirements/Test Team will categorize and prioritize the issues raised by the users. The UAT participant comments will be reviewed during the scope definition phase of future releases of RASS.

5.3.5.2 Setup

RASS UAT will be conducted in the RASS String Test environment. Please refer to Section 5.1.5.2 Setup for more information on the setup of the integrated environment.

5.3.5.3 Initialization

UAT will be initialized when the test environment has been refreshed with a partial copy of the production data set and a copy of the new system source code components. Initialization indicates when a new pass of testing may commence.

The RASS Testing team will initialize the test environment between passes by refreshing the String Test database. The PVCS Version Manager will be used to migrate new system source code components to the String Test environment for the UAT.

5.3.5.4 Preparation

The following will occur in preparation for User Acceptance Test.

- The RASS Release 8.4.0.0 Validation, Verification, and Testing Plan (HUD SDM Design Phase), including the Test Conditions and Expected Results document will be reviewed and delivered.
- A list of UAT instructions and test data will be created and reviewed.
- Testers will have a set of internal and external user ids.
- The String Test, System Test and Integration Test will be complete.
- The String Test environment will have a newly refreshed partial copy of production data with the system components from PVCS Version Manager.

5.3.5.5 Termination

The UAT is successful if all actual results meet expected results. Any comments or feedback from the UAT will be considered for implementation in future releases. User Acceptance Testing will end when the list of instructions has been completed successfully, and the UAT participants' comments have been recorded. The UAT test results will be compiled and given to project management for signoff.